An Agent-based framework for mitigating hazardous materials transport risk

Hassan Kanj
and Jean-Marie Flaus
Univ.Grenoble Alpes, G-SCOP, F-38000 Grenoble, France
CNRS, G-SCOP, F-38000 Grenoble, France
Email: hassan.kanj@grenoble-inp.fr

***Abstract*—Dangerous goods transportation (DGT) represents technological and environmental risks for exposed populations, infrastructures and environment. Historical evidence has shown that road-accidents in DGT can lead to various potential consequences characterized by fatalities, injuries, evacuation, property damage, environmental degradation, and traffic disruption. Due to the importance of these products in everyday civil life activities and the increase in demand for these materials, developing tools for risk analysis and mitigation becomes a strategic goal in particular in those countries, like France, in which the majority of goods are transported by road. Based on the complexity of the dangerous goods transportation system DGTS and its related risk (factors that characterized risks are time dependent as traffic conditions, weather conditions, incident probability and population exposure), this analysis can only be made via simulation. This paper describes a generic approach to use agent-based modeling, an interesting approach to modeling systems comprised of autonomous and interacting agents, for risk analysis. It presents a novel generic model facet for representing risk analysis and fault tree propagation in an agent model, where the goal is to analyze the risk related to a system and to simulate its behavior in normal and degraded mode by using multi-agents systems. This approach is used to analyze the risks related to dangerous goods transportation and to minimize these risks by using agent-based model (identifying the best road that having the minimum risk level for transport).**

I. Introduction

Every day, dangerous goods (DG) are transported in different modalities from one or more origins to their destinations, all over the word, where people needs DG to live, to work, but also to find out new frontiers. Every country needs DG for everyday civil life activities: for example to use energy, to transport goods and passengers, or simply to conduct a healthy and safe life [1].

Dangerous goods transportation (DGT) includes all goods - liquids, gasses, and solids - that include radioactive, flammable, explosive, corrosive, oxidizing, asphyxiating, biohazardous, toxic, pathogenic, or allergenic materials. All substances that induce severe risk for health, that can harm people, environment and surrounding properties, or other living organisms, are characterized as DG [2]. The severity of consequences related to road-accidents in DGT and the large number of lorries transported DG by day requires developing tools for risk analysis and mitigation, where the risk is defined as product event frequency and its consequences. These risks might be characterized by different aspects as type and quantity of dangerous goods, vehicle and road characteristics, traffic and weather conditions and population density. Some of these aspects are time dependent (traffic and weather conditions, population density), which implies the complexity of such analysis and encourages the use of simulation model. A simulation model may be considered as a set of rules (e.g. equations, flowcharts, state machines, cellular automata) that define how the system being modeled will change in the future, given its present state. Simulation is the process of model execution that takes the model through (discrete or continuous) state changes over time.

Many approaches may be used for Simulation modeling, such as:

- System Dynamics (SD), developed by the electrical engineer Jay W.Forrester in the 1950s [3]. It is mainly continuous and is characterized by a high abstraction level, low details and a strategic level;
- Discrete Event (DE), which roots back to 1960s when Geoffrey Gordon conceived and evolved the idea for General purpose simulation system (GPSS) and brought about its IBM implementations [4]. This model is mainly discrete and characterized by a middle abstraction level, medium details and a tactical level [5];
- Agent Based (AB), which is known by many names. ABM (agent-based modeling), ABS (agent-based systems or simulation), and IBM (individual-based modeling) are all widely used acronyms. These models are essentially decentralized and preferred for complex systems. They can range from high to low abstraction levels [5].

As the analyzed system is time dependent, complex in terms of multiplicity of units, decentralization of decision making, and number of relationships between its components, then authors will use an ABS.

In this paper, a generic model facet for representing risk analysis and fault propagation in an agent model has been proposed. This facet can be built in a systematic manner from model based risk analysis and is made of the following elements:

- a set of behavioral modes and an associated activity model represented as an activity UML model.
- a set of events and a dysfunctional model represented by a bow tie model.

- a set of transition rules to describe the interaction between these models

The remainder of this paper is organized as follows. Section II gives a basic representation of the multi-agents systems (MAS) and define the agents, their attributes and their relations. Section III presents the proposed model, and explain how to adapt a multi-agents system to risk analysis. Section IV illustrates a proposed algorithm to mitigating risks related to a DGT and shows for each agent class an agent-based model. Finally, Section V concludes this paper.

II. Multi-agent system (MAS)

A. Definition of an agent

An agent is an independent entity with precise boundaries and specific goals that exhibits autonomous behavior and has both sensorial and communicational capabilities. It may have incomplete information about its surroundings and limited capacity to influence others.

In literature, a variety of definitions exist to describe what an agent is ([6], [7]). One example, described in [7], defines an agent as a physical or virtual entity :

- which is able to act in an environment and can eventually reproduce,
- that can communicate directly with other agents,
- which is driven by a set of trends (in the form of individual goals or function of satisfaction, even survival), it seeks to optimize,
- which is able to receive (to a limited extent) its environment,
- which may or may not have only a partial representation of this environment,
- which is expert and provides services and has its own resources,
- whose behavior tends to meet its objectives, taking into account the resources and expertise available to it, and according to its perception, its representations, and communications it receives.

B. Properties of an agent

The concept of agents is closely related to the object oriented approach used in modern programming languages such as Java or C++. Thereby an object is defined by its states and behavior, whereas agents can be seen as objects with more extended capabilities (e.g., rules of behavior, autonomy, co-operation (e.g., perception, action, communication), mobility, memory, learning ability, among others. Cooperation, which is considered as a core capability of an agent, comprises, e.g., perception and action (interaction) and communication.

C. Definition of MAS

The multi-agent system is a software technology in great demand to model and simulate the dynamic behavior of complex and decentralized systems, known as $Intelligentsystems$. Historically, these systems are positioned at the intersection of programming (software), artificial intelligence (decision-making autonomy) and distributed systems (decentralization).

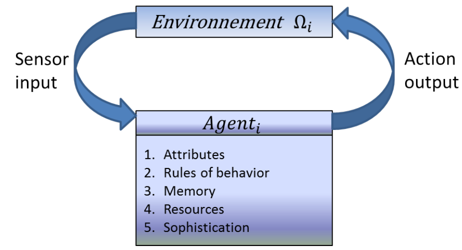

Fig. 1. An Agent is in a mutual communication with its environment

Agent-based modeling (ABM) is is a micro-simulation tool that recreates in a virtual context the interactions and behaviors of a set of autonomous agents and their environment. This tool considers that any system is made up of a set of entities agents that interact amongst themselves and with an environment that supports their very existence [8]. ABM is also considered as upcoming approach in complex systems science to model structures comprising autonomous and interacting elements. Some scientists even denote this computer simulation based modeling approach ”A New Kind of Science” and argue, that besides deduction and induction, ABM and simulation is a third way of doing science.

The basic idea of ABM is to model only the units - called agents - of a specific system and to simulate their interplay in order to derive and analyze the total system behavior.

Building an agent based simulation is composed of three parts:

- Define the set of agents and the environment which contains all agents: identify for each agent a set of attributes, behavior modes, and resources, and for the environment, its characteristics, responsibility and functions;
- precise for each functions, its characteristics and attributes;
- Specify the interactions in the system.

D. Environment

It provides the physical support for the agents live-ability and interactions [9]. The environment of an agent j (Ω_j) is defined as the set of all elements or objects exterior to j. These elements define a common space of action for agents. Figure 1 shows an agent that takes sensory inputs from its environment, and produces as output actions that affect it.

E. Interactions between agents

Two types of communication can be observed for an agent (see figure 2):

- agent-agent (direct communication) : is carried in an intentional way by sending messages to one or more well defined recipients;

- environment-agent or indirect communication: is carried either through the environment, not intentional action, leaving traces or signals, or through a chalkboard (intentional action) filing and reading information filed in a shared data area. In this type of communication, the recipients are not defined, they are all agents that are in the vicinity of the agent sender.

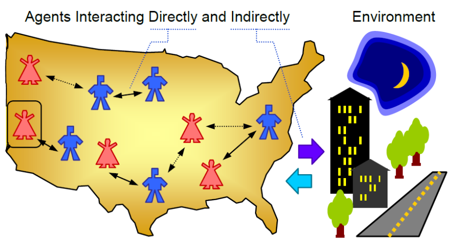

Fig. 2. Agents interactions [5]

III. ABM FOR RISK ANALYSIS

To adapt the model-based agents to risk analysis, the authors proposes some additional characteristics to a classical agent as:

1) behavioral modes: described by a list of successive blocks
2) failures modes: for expressing the risk and dysfunction analysis of system-agents
3) relationships between failure modes and behavioral modes of system-agents.

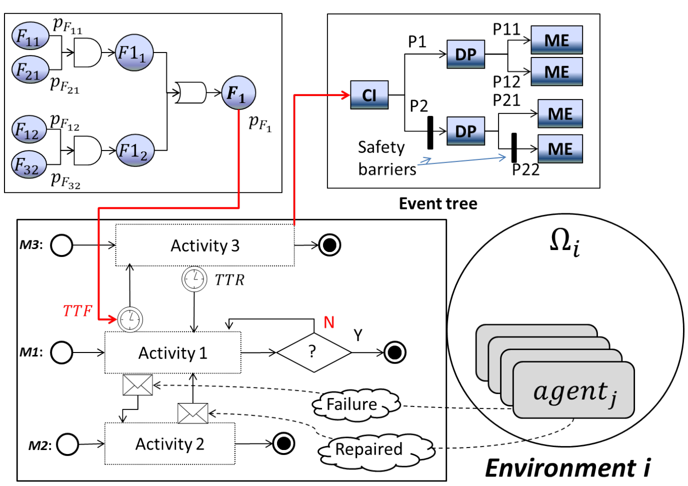

Fig. 3. Generic Agent-oriented risk model (agent i)

Then they are used the Function interaction structure modeling tool to model a multi-agent system, The structure of each agent is composed of three views:

1) Structural view
2) Dysfunctional view
3) Behavioral View

A. Structural view

This part describes the structure of an agent, by showing its relations with its environment (other agents) and the characteristics of each of them.This basic view, allows us to enter into the dysfunctional or behavioral view.

1) Structure of the agent: The basic element is the agent, seen as an intelligent entity capable of interacting with its environment and with other agents, execute tasks described by an organized set of activities that use resources or agents (personnel, equipment and machinery, ...) to transform from input into output agents. It is described internally by:

- Structural elements that are defined with a list of properties of accumulated resources and behavioral rules.
  The property list of an agent may contain:
  - a type, that defines the class under which the agent belongs;
  - a name, which specifies the name to identify the agent from the other agents in its environment;
  - a list of variables or characteristics: for each characteristic, define a name, a type and a domain of validity.

  The rules of behavior vary by agent. They include:
  - The rules of sophistication
  - The cognitive load
  - The internal models of the external world
  - The memory used
- The $functional\ elements$ (FE), called $activities$ or $tasks$ for an agent. An activity can be either active or inactive at a given time t. Its behavior is characterized by a set of variables, and eventually by a behavioral model describing the temporal evolution.

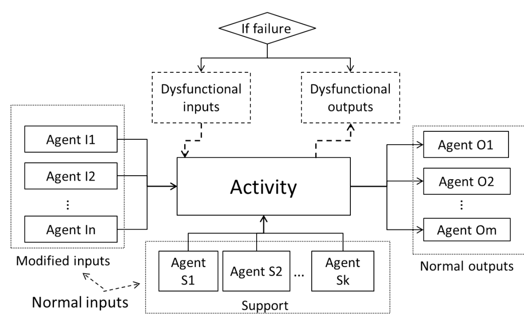

Fig. 4. Relations activity

2) Links between activities and resources: Relationships can take place between the functional elements of an agent and other agents in its environment, which is necessary to perform an activity (see figure 4). These relationships are of three types:

- input relations : sometimes a number of agents is requested in entering an activity. These agents are either consumed or provide information that is used;
- outputs relations: other agents are either generated or their status are changed, it is the case of a material

agent that is produced, or object that is assembled agent;

- relationships of use: in this case, the agents are considered supports for the function. For example, to advance, a truck agent use a driver agent.

When an activity failure mode becomes active, the activity takes dysfunctional inputs and provides dysfunctional outputs.

B. Dysfunctional view

Dysfunctional view describes the propagation of failure between model elements. For example, a failure of an input or support resources for an activity may cause the failure of the activity or may produce an output resources in a degraded state.

Two categories of failure sources can be observed:

1) Failure related to the activity: a failure of an input or support agent for an activity cause the failure of the activity;
2) Exterior failure related to others agents or to the environment: where the agent receive a message form its environment which indicates that a failure occurred (it contains a description of the failure), this message is stored in its inbox.

Each activity-failure mode is described by:

- a name;
- a set of events, causes for this mode, connected using AND and OR gates represented in the fault tree and a set of events ,consequences of the failure, representing in the event tree;
- a time t representing the time to failure (TTF) of this mode (for activity failure) .
- and a time TTR which presents the needed time to repair the failure (once, current time begin equals TTF + TTR for a failure, this failure become inactive).

And each exterior failure is described by:

- a name;
- a Boolean value: when an agent receive a failure-message from its environment, which contains the id of the failure, this failure become true and when he receive a reparation message with the id of failure (repaired), the failure value becomes false.

Failure mode is denoted by $fmode_i$. The normal mode of an activity is defined as :

$$ok = \bigwedge_i \neg_{fmode_i} \quad (1)$$

Once, this view is built, we can define the set of failure modes for each agent's activity in the system. Afterward, we add the set of exterior failure in order to obtain for each agent a set of failure mode.

C. Behavioral view

This view describes the dynamic behavior of the agent. Once, we have this view for all agent, it is possible to simulate the operation of the system, in normal or degraded modes. In addition, some elements of this view can be exploited for diagnosis or prognosis [10].

1) Activity Diagram: The behavior of each agent is described by an activity block diagram. For an agent, many behavioral modes are possible, however, at a precise time t, only one mode is active while the others are inactive. A behavioral mode represent a program for the agent with many input parameters. A mode has a current state (active, inactive) which is updated in each step of the simulation. It may contain many types of blocks:

- A start event block: This is the first block executed by an agent, it indicates the existence of a new agent in the environment;
- An end event block: This is the last block executed by an agent, it indicates the end of life of an agent;
- An activity block called $transformation\ activity$: this type of activity involves interaction between an agent and its environment, and can lead to changes in the characteristics of the agents and the environment. It is defined by:
  - a transformation relation :
  $$\sum n_i me_i|\phi_i.. \rightarrow \overset{n_s me_s|\phi_s}{\longrightarrow} \sum n_o me_o|\phi_o\{m_i\}, \quad (2)$$
  where me_i is an input model element, required in number n_i and which must satisfy the condition ϕ, which a logical relation expressed with respect to the variables, the attributes and the events available in the scope. me_o is an output and me_s is a model element representing a support;
  - a positive integer duration δ representing the time
  $$t = \delta.T_s \quad (3)$$
  and/or a set of final conditions;
  - a set of input/ outputs actions;

  It is also possible for an interaction of this kind to create new agents in the environment or kill existing agents. This type of activity is sometimes complex. We can decompose it into a list of simple activities.
- A gateway block: it used for testing the value of a Boolean equation. Usually, this block is followed by 2 blocks, the first block is active when the equation is true and the second is active when it is false.
- A block with many successors: this type of block is used to represent the position of an agent that executes more than one block after the execution of a block. In this case, once the execution of the block is completed, all successor blocks are activated .
- A block with many predecessors: this type of block is used to represent the position of a block that is active only when several blocks are already executed.

When a behavioral mode for an agent becomes active, we execute its first block (usually the start event). We then identify its successor block and check if it is an activity block. If not, we propagate the activation. If the next block is of type gateway we test its expression to determine its successors. When a block has an activity block as successors, we test all input and support elements (agents) of this activity block. If they are all available, this activity block becomes active running:

- we execute its input actions and make time t = current time after we execute its task when t + duration of block is less then current time;
- Once, current time = t + duration of block, we execute the output actions of the block, we destroy its input, we release its support and output agents, and this block becomes inactive.

A behavioral mode of an agent is finished when we execute its final block (end event).

D. Relations between dysfunctional and behavioral views

During the construction of a multi-agent model and after defining the set of agents in the system, we define for each agent:

1) a behavioral mode called **nominal behavioral mode**, is defined (mode m1 in the figure 3);
2) a set of failure modes associated for each activity in its nominal mode plus the set of exterior failures;
3) a set of failures modes which contains the set of failure modes of all its activities;

Then, for each combination of failure mode for an agent, we define a corresponding behavioral modes (see Figure 3) (for example: when f1 becomes active, the behavioral mode for the agent change from M1 to M3 and when it receive a failure message from its environment, its behavioral mode becomes M2).

Now, for each agent in the environment, they have:

- a set of variables;
- a set of behavioral modes: with one representing the normal mode (defined by default) and the others representing the dysfunctional modes;
- a set of failure modes;
- a set of relations between the dysfunctional and behavioral mode;

The number of behavioral modes of an agent may be up to 2^n where n: is the number of its failure modes. The number is, however, usually much less as a same behavior mode is associated to several failure modes.

E. Transition rule from a mode to another

At each step in the simulation, a test is carried out on the entire agent failure modes (activity failure and received messages in its inbox). This allows us to identify which failure modes should be true in the next step. Based on the value of each failure mode, we can determine if the current behavioral mode remains active for the next step or replace it by another mode. If it is to be replaced, we must stop all running activities that correspond to the old mode and activate the first block in the novel mode.

Figure 3 shows an agent that has a normal behavioral mode m1. In this mode, there is an activity $Activity1$ which has two failure modes: the first is related to the activity and represented by a fault tree (with time to failure equals TTF) and the other one is exterior failure mode. Once, a failure mode become active, if it is the activity failure mode, the behavioral mode of the agent becomes m3 and Activity1 becomes inactive, it is possible that M1 returns active when the current time = TTF + TTR. Else, if it is an exterior failure, the behavioral mode of the agent becomes M2 and Activity 1 becomes inactive. It is also possible that the agent behave according to m1 when the agent receive a message $repaired$. So finally, for each failure mode, there is a corresponding behavioral mode.

IV. ABM FOR OPTIMIZING AND MITIGATING DGTR

The use of agent technology for the study of freight transport systems is relatively recent, with the first publications dating back to the 1990s [11]. An early attempt to use ABM principles in solving transport-related problems was conducted by [12]. The purpose of the work was to minimize the expected total risk as defined by the multiplication of the accident probability and its severity (number of exposed people). The reviewed literature and the analysis of the case study provided information about the agents involved in the identification of the best road. A total of seven agents were identified: truck (i), driver, provider weather conditions, provider Segment characteristics, provider traffic conditions, provider population and provider dangerous goods characteristics.

The authors supposes that minimizing these risks can be achieved in two steps:

1) identify the best road (prior optimization): precise which road has the minimum level of risk, by applying, for each of possible road, the following algorithm:
   - simulate an advancement of the truck for a time delta t, and identify its position after the advancement;
   - calculate the probability of an accident during the advancement according to the dynamic parameters: weather and traffic conditions;
   - calculate the severity of the accident: evaluate its severity (number of exposed people);
   - return to step 1 until reach the destination.

   The expected total risk for a road is defined as:

$$Risque = P_1 \times G_1 + \sum_{j=2}^{n-1} \prod_{i=1}^{j-1} (1 - P_i) \times P_j \times G_j \quad (4)$$

where:
P_i: probability of an accident that occurred during advancement number i;
G_j: severity of the accident that occurred during advancement number i;
and N: number of advancement.

As Results, there are for each road the risk level and the expected travel time, and the best road is that has the minimum risk level. Figure (5) presents the behavioral mode related to the truck agent, where the authors considers that at each advancement an accident occurs, the time to reparation TTR is equals 0, figure 6 shows the relations of the moving activity and figure 7 illustrates the relations of the activity calculating the severity. It is important to notice that more than one road can be best road and a change in the departure time may change the best road). This step is also named prior optimization because the optimal road is chosen before the travel begins.

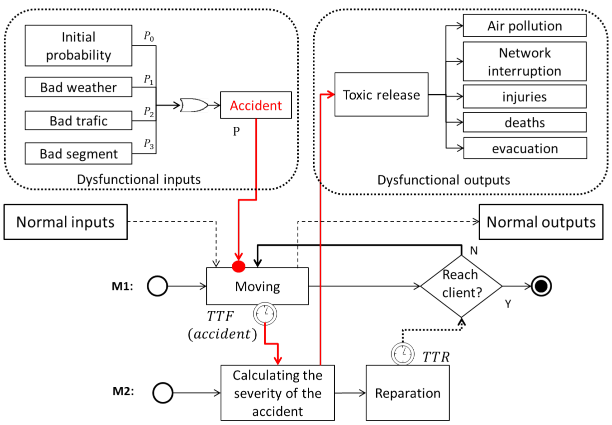

Fig. 5. Agent based model for truck

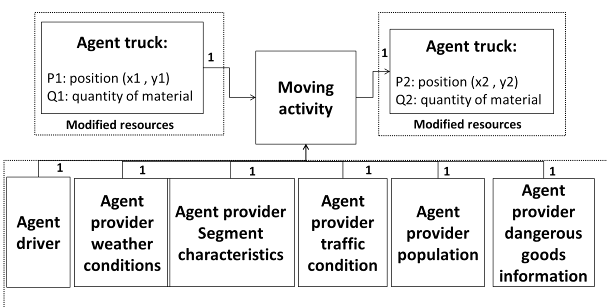

Fig. 6. Relations of Moving activity

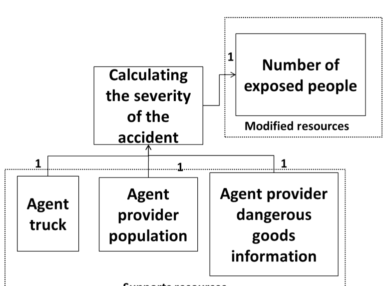

Fig. 7. Calculating the severity of the accident

2) minimize effects: after identifying the best road to travel, the travels begins. Here, some agents are added in the simulation as agent operators (o), control center (k), secour (s) and vehicle (j). The interaction and communication between agents appears explicitly. The authors suppose that an agent A can communicate with other agent B if the distance between them is less then or equals to a distance d, and during its advancement, a truck is always in communication with at least one control center in which the agent control center have a complete information of the position of the truck and its content at each time t.
At first, all agent behaves in their normal behavioral modes, vehicle and truck move in a map according to their behavioral modes. During the advancement, an accident can occurs on a truck or a vehicle. When an accident occur on a truck, its behavioral mode change from m1 to m2 and it send directly a message to the control center and the nearest operator and send another broadcasts message to all agents (vehicles or trucks) in its area to inform them about the accident (see figure 9). Once, the control center receives the message, it change its behavioral modes and sends a notification message to the closest secour which contains information about the incident and its position (see figure 12). In turn, secour change its behavioral modes and moves to the position to evacuate people from the dangerous area to another safe (isolate) area and operator change its behavioral modes and moves to the position to repair the failure (see figure 10). When, operator finishes its tasks, it returns to its old behavioral mode and it sends a message repaired to the truck (see figure 8) and when secour evacuate all people, it returns to its old behavioral mode and sends a message Ok to the control center which in turn returns to its old behavioral modes. When a vehicle or a truck receive the broadcast message, it change its behavioral mode from m1 to m2 and its treat the message: if its is an old message or a message already received , it do anything (to avoid recursive or inappropriate messages) else it broadcasts the message in its environment and it test if the position of the accident is in its trajectory, it changes its direction if not, it continues normally.
In the other case, when a failure occur on a vehicle (figure 11), it send a message to the nearest operator to repair its failure and sends another broadcast message in its environment. Once, the operator receive the message, it move towards the vehicle (the authors considers that the time in route is equals 0). When it finish the reparation, it sends a message repaired to the vehicle which continues its trajectory. Figure 13 shows the relations of the activity advancement truck, and figure 14 illustrates the relations of advancement vehicle activity.
The autonomy and the intelligence of agents provides many benefits to the system as: when a truck or vehicle receives a message, they broadcasts it if and only if it is an important message, this methodology minimize the size of sending message and the used memory, also these agents are able to identify the best route when an accident occurs in their trajectory. And finally, exchanged messages between agents lead to a rapid intervention by the agents secours and operators in the case of accident which can minimize the effects

related to it.

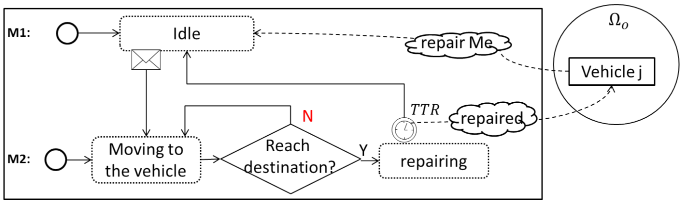

Fig. 8. Agent-based model of the operators

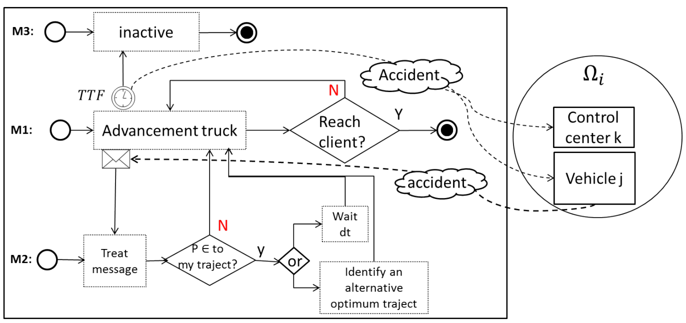

Fig. 9. Agent-based model of the trucks

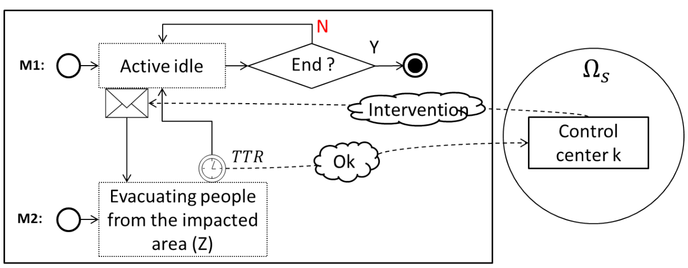

Fig. 10. Agent-based model of the secours

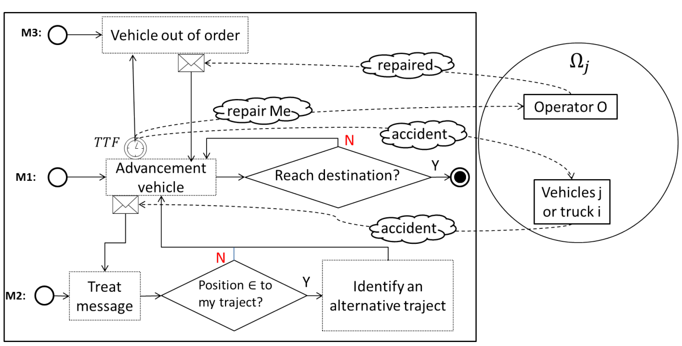

Fig. 11. Agent-based model of the vehicles

For shipping dangerous goods from Grenoble to Lyon, many roads can be used. In this example, the authors take count two roads (road 1 and road 2, road 2 is longer than road 1). To identify the best road between them, they are used the algorithm cited in section IV. Figure 15 illustrates for each road the number of people affected at each advancement (4 minutes). It appears that between time 0 and 44, the number of people impacted at the road 1 is bigger than that impacted at road 2 and from time 44 to 48, the number of impacted people at road 2 is the biggest. Figure 16 shows at each advancement

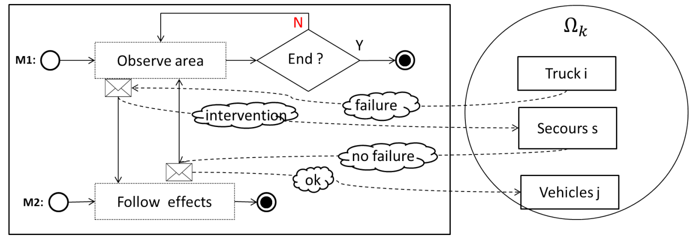

Fig. 12. Agent-based model of the control center

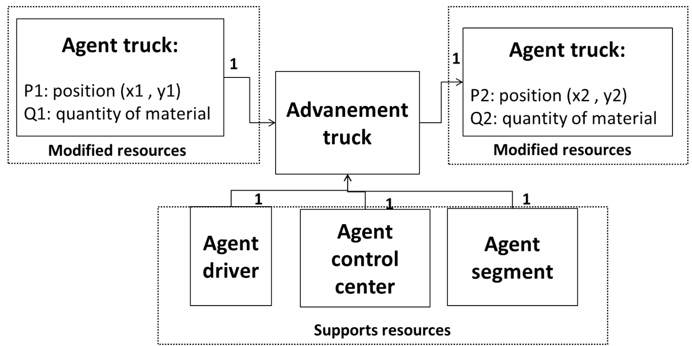

Fig. 13. Relation of advancement truck activity

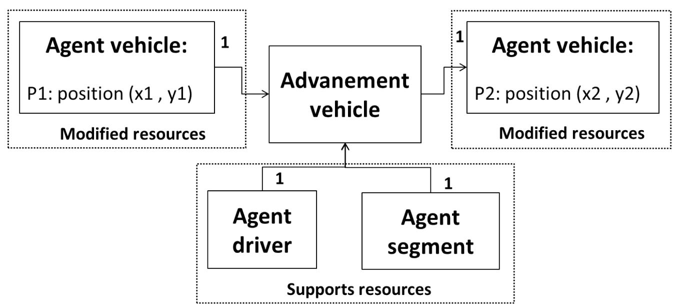

Fig. 14. Relation of advancement vehicle activity

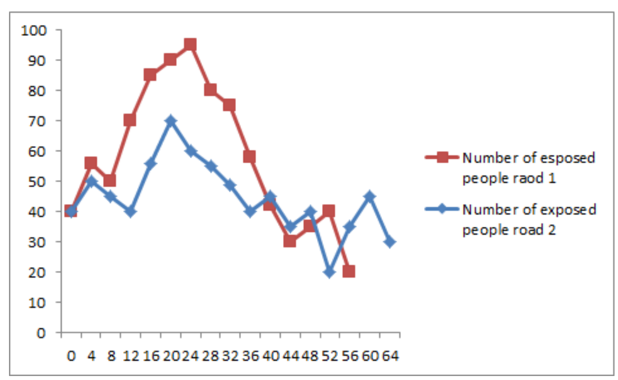

Fig. 15. Number of exposed people

the hazardous areas (two hazardous areas can be observed: area of lethal effects and area of irreversible effects).

After computing the expected total risk for these roads, it appears that road 2 has the minimum level of risk and it is used in the transport.

The next step after identifying the best road is to start

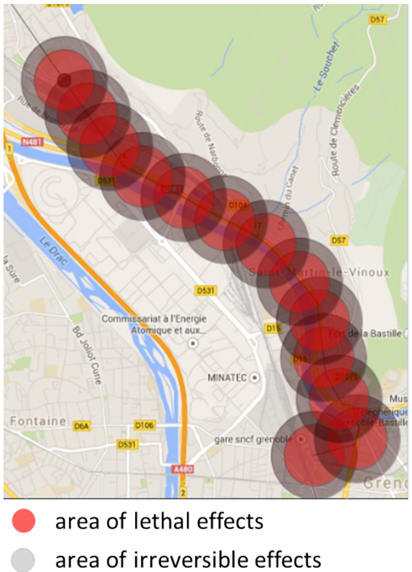

Fig. 16. Affected areas

the travels. Now, the second step in the section IV is used to minimize and manage risks.

V. CONCLUSIONS

In this paper, a new approach for risk analysis based on multi-agents have been presented. For each agent in the system, there is generic model which contains a set of activities, attributes, failure modes and behavioral modes. The main interest of this meta model is that it allows the representation of risk analysis and dynamical behavior in a coherent manner, which can be used to simulate the behavior of a system in normal or degraded conditions. This was used to evaluate and minimize the level of risk related to dangerous goods transportation throughout an example of a moving of a truck agent loaded with hazardous materials. As future goals, for the same example illustrated in this paper, it is possible to treat the existence of failure agents, which sent wrong messages and how the others agents can discover them. Also this model can be applied to a system containing a large number of active agents as evacuation system.

REFERENCES

[1] A. M. Tomasoni, "Modèles et méthodes d'évaluation et de gestion des risques appliquées aux systèmes de transport de marchandises dangereuses (tmd), reposant sur les nouvelles technologies de l'information et de la communication (ntic)," Ph.D. dissertation, Paris, ENMP, 2010.

[2] K. G. Zografos, G. M. Vasilakis, and I. M. Giannouli, "Methodological framework for developing decision support systems (dss) for hazardous materials emergency response operations," *Journal of hazardous materials*, vol. 71, no. 1, pp. 503–521, 2000.

[3] J. W. Forrester, "Industrial dynamics: a major breakthrough for decision makers," *Harvard business review*, vol. 36, no. 4, pp. 37–66, 1958.

[4] G. Gordon, "A general purpose systems simulation program," in *Proceedings of the December , eastern joint computer conference: computers-key to total systems control*. ACM, 1961, pp. 87–104.

[5] A. Borshchev and A. Filippov, "From system dynamics and discrete event to practical agent based modeling: reasons, techniques, tools," in *Proceedings of the 22nd international conference of the system dynamics society*, vol. 22, 2004.

[6] M. Wooldridge and N. R. Jennings, "Intelligent agents: Theory and practice," *The knowledge engineering review*, vol. 10, no. 02, pp. 115–152, 1995.

[7] J. Ferber and J.-F. Perrot, *Les systèmes multi-agents: vers une intelligence collective*. InterEditions, 1995.

[8] E. Bonabeau, "Agent-based modeling: Methods and techniques for simulating human systems," *Proceedings of the National Academy of Sciences*, vol. 99, no. suppl 3, pp. 7280–7287, 2002.

[9] M. J. North and C. M. Macal, *Managing business complexity: discovering strategic solutions with agent-based modeling and simulation*. Oxford University Press, 2007.

[10] J.-M. Flaus, *Risk Analysis: Socio-technical and Industrial Systems*. John Wiley & Sons, 2013.

[11] L. Tavasszy, "Freight modeling: An overview of international experiences (with discussion)," in *Transportation Research Board Conference Proceedings*, no. 40, 2008, pp. 47–55.

[12] K. Fischer, N. Kuhn, H. Müller, J. P. Müller, and M. Pischel, "Sophisticated and distributed: The transportation domain," in *From Reaction to Cognition*. Springer, 1995, pp. 122–138.